

**The Claims**

1. (Currently Amended) A method for providing a graphical user interface in a High Performance Computing (HPC) environment comprises comprising:

collecting dynamic status information on each of at least a subset of a plurality of HPC nodes, each node comprising an integrated fabric a switching fabric integrated to a card and at least two processors integrated to the card;

generating a plurality of graphical elements based, at least in part, on the collected information that convey at least some of the dynamic status information; and

presenting at least a portion of the graphical elements communicating at least some of the graphical elements for presentation to a user.

2. (Currently Amended) The method of Claim 1, wherein the collected dynamic status information comprising physical data on each HPC node on a node indicates a physical status of the node.

3. (Currently Amended) The method of Claim 2, wherein the physical data comprising physical status of the node comprises one or more of the following:

processor utilization;

memory utilization;

physical location;

IP address; and or

bandwidth.

4. (Currently Amended) The method of Claim 1, wherein one of at least some of the graphical elements comprising collectively enable a view of a topology of the plurality of HPC at least the subset of the nodes, the switching fabrics of the nodes enabling the topology enabled by the integrated fabric of each node.

5. (Currently Amended) The method of Claim 4, further comprising:  
receiving a job submission from the user, the job submission comprising at least one parameter;  
communicating the job submission to a job scheduler for dynamic allocation of a ~~second particular~~ subset of the ~~plurality of HPC nodes~~ to the job submission; and  
updating the view of the topology based on the dynamic allocation of the ~~second particular~~ subset.

6. (Currently Amended) The method of Claim 5, further comprising:  
communicating an interactive command to the job scheduler ~~for an~~ to increase ~~in a~~ size of the ~~second particular~~ subset; and  
updating the view of the topology based on ~~dynamic allocation of the increased size~~ the increase.

7. (Currently Amended) The method of Claim 4, further comprising:  
receiving a notification of a failure of one of the ~~plurality of HPC nodes~~; and  
updating the view of the topology based on the notification.

8. (Currently Amended) ~~A Graphical User Interface (GUI) in a High Performance Computing (HPC) environment Software encoded in one or more computer-readable tangible media and when executed operable to:~~  
collect dynamic status information on each of at least a subset of a plurality of HPC nodes, each node comprising an integrated fabrie a switching fabric integrated to a card and at least two processors integrated to the card;

generate a plurality of graphical elements ~~based, at least in part, on the collected information~~ that convey at least some of the dynamic status information; and

~~present at least a portion of the graphical elements~~ communicate at least some of the graphical elements for presentation to a user.

9. (Currently Amended) The GUI software of Claim 8, wherein the collected dynamic status information comprising physical data on each HPC node on a node indicates a physical status of the node.

10. (Currently Amended) The GUI software of Claim 9, wherein the physical data comprising physical status of the node comprises one or more of the following:

processor utilization;  
memory utilization;  
physical location;  
IP address; and or  
bandwidth.

11. (Currently Amended) The GUI software of Claim 8, wherein one of at least some of the graphical elements comprising collectively enable a view of a topology of the plurality of HPC at least the subset of the nodes, the switching fabrics of the nodes enabling the topology enabled by the integrated fabric of each node.

12. (Currently Amended) The GUI software of Claim 11, further operable to:  
receive a job submission from the user, the job submission comprising at least one parameter;  
communicate the job submission to a job scheduler for dynamic allocation of a second particular subset of the plurality of HPC nodes to the job submission; and  
update the view of the topology based on the dynamic allocation of the second particular subset.

13. (Currently Amended) The GUI software of Claim 12, further operable to:  
communicate an interactive command to the job scheduler ~~for an~~ to increase ~~in~~ a size of  
the second particular subset; and  
update the view of the topology based on ~~dynamic allocation of the increased size~~ the  
increase.

14. (Currently Amended) The GUI software of Claim 11, further operable to:  
receive a notification of a failure of one of the ~~plurality of HPC~~ nodes; and  
update the view of the topology based on the notification.

15. (Currently Amended) A system ~~for providing a graphical user interface in a High Performance Computing (HPC) environment~~ comprises comprising:  
~~a plurality of HPC nodes, each node comprising an integrated fabric~~ a switching fabric  
integrated to a card and at least two processors integrated to the card; and  
a client operable to:  
collect dynamic status information on ~~at least a subset of the plurality~~ each of  
HPC at least a subset of the nodes;  
generate a plurality of graphical elements ~~based, at least in part, on the collected~~  
information that convey at least some of the dynamic status information; and  
~~present at least a portion of the graphical elements~~ communicate at least some of  
the graphical elements for presentation to a user.

16. (Currently Amended) The system of Claim 15, wherein the collected dynamic  
status information comprising physical data on each HPC node on a node indicates a physical  
status of the node.

17. (Currently Amended) The system of Claim 16, wherein the physical data comprising physical status of the node comprises one or more of the following:

processor utilization;  
memory utilization;  
physical location;  
IP address; and or  
bandwidth.

18. (Currently Amended) The system of Claim 15, wherein one of at least some of the graphical elements comprising collectively enable a view of a topology of the plurality of HPC at least the subset of the nodes, the switching fabrics of the nodes enabling the topology enabled by the integrated fabric of each node.

19. (Currently Amended) The system of Claim 18, the client further operable to:  
receive a job submission from the user, the job submission comprising at least one parameter;  
communicate the job submission to a job scheduler for dynamic allocation of a second particular subset of the plurality of HPC nodes to the job submission; and  
update the view of the topology based on the dynamic allocation of the second particular subset.

20. (Currently Amended) The system of Claim 19, the client further operable to:  
communicate an interactive command to the job scheduler for an to increase in a size of the second particular subset; and  
update the view of the topology based on dynamic allocation of the increased size the increase.

21. (Currently Amended) The system of Claim 18, the client further operable to: receive a notification of a failure of one of the ~~plurality of HPC~~ nodes; and update the view of the topology based on the notification.
22. (New) The method of Claim 1, wherein each card is a motherboard.
23. (New) The GUI of Claim 8, wherein each card is a motherboard.
24. (New) The system of Claim 15, wherein each card is a motherboard.

25. (New) A method comprising:  
collecting dynamic status information on each of at least a subset of a plurality of nodes,  
each node comprising:

at least two first processors integrated to a first card and operable to communicate  
with each other via a direct link between them; and

a first switch integrated to the first card, the first processors communicably  
coupled to the first switch, the first switch operable to communicably couple the first  
processors to six or more second cards each comprising at least two second processors  
integrated to the second card and a second switch integrated to the second card operable  
to communicably couple the second processors to the first card and at least five third  
cards each comprising at least two third processors integrated to the third card and a third  
switch integrated to the third card;

the first processors being operable to communicate with particular second  
processors on a particular second card via the first switch and the second switch on the  
particular second card;

the first processors being operable to communicate with particular third  
processors on a particular third card via the first switch, a particular second switch on a  
particular second card between the first card and the particular third card, and the third  
switch on the particular third card without communicating via either second processor on  
the particular second card;

generating a plurality of graphical elements that convey at least some of the dynamic  
status information; and

communicating at least some of the graphical elements for presentation to a user.

26. (New) Software encoded in one or more computer-readable tangible media and when executed operable to:

collect dynamic status information on each of at least a subset of a plurality of nodes, each node comprising:

at least two first processors integrated to a first card and operable to communicate with each other via a direct link between them; and

a first switch integrated to the first card, the first processors communicably coupled to the first switch, the first switch operable to communicably couple the first processors to six or more second cards each comprising at least two second processors integrated to the second card and a second switch integrated to the second card operable to communicably couple the second processors to the first card and at least five third cards each comprising at least two third processors integrated to the third card and a third switch integrated to the third card;

the first processors being operable to communicate with particular second processors on a particular second card via the first switch and the second switch on the particular second card;

the first processors being operable to communicate with particular third processors on a particular third card via the first switch, a particular second switch on a particular second card between the first card and the particular third card, and the third switch on the particular third card without communicating via either second processor on the particular second card;

generate a plurality of graphical elements that convey at least some of the dynamic status information; and

communicate at least some of the graphical elements for presentation to a user.

27. (New) A system comprising:

a plurality of nodes, each node comprising:

at least two first processors integrated to a first card and operable to communicate with each other via a direct link between them; and

a first switch integrated to the first card, the first processors communicably coupled to the first switch, the first switch operable to communicably couple the first processors to six or more second cards each comprising at least two second processors integrated to the second card and a second switch integrated to the second card operable to communicably couple the second processors to the first card and at least five third cards each comprising at least two third processors integrated to the third card and a third switch integrated to the third card;

the first processors being operable to communicate with particular second processors on a particular second card via the first switch and the second switch on the particular second card;

the first processors being operable to communicate with particular third processors on a particular third card via the first switch, a particular second switch on a particular second card between the first card and the particular third card, and the third switch on the particular third card without communicating via either second processor on the particular second card; and

a client operable to:

collect dynamic status information on each of at least a subset of the nodes;

generate a plurality of graphical elements that convey at least some of the dynamic status information; and

communicate at least some of the graphical elements for presentation to a user.